MOS TECHNOLOGY, INC.
VALLEY FORGE CORPORATE CENTER (215) 666-7950
950 RITTENHOUSE ROAD, NORRISTOWN, PA. 19401

PRELIMINARY

DATA

SHEET

MAY, 1976

# MCS6500 MICROPROCESSORS

## The MCS6500 Microprocessor Family Concept ----

The MCS6500 Series Microprocessors represent the first totally software compatible microprocessor family. This family of products includes a range of software compatible microprocessors which provide a selection of addressable memory range, interrupt input options and on-chip clock osscillators and drivers. All of the microprocessors in the MCS6500 group are software compatible within the group and are bus compatible with the M6800 product offering.

The family includes five microprocessors with on-board clock oscillators and drivers and four microprocessors driven by external clocks. The on-chip clock versions are aimed at high performance, low cost applications where single phase inputs, crystal or RC inputs provide the time base. The external clock versions are geared for the multi processor system applications where maximum timing control is mandatory. All versions of the microprocessors are available in 1 MHz and 2 MHz ("A" suffix on product numbers) maximum operating frequencies.

## Features of the MCS6500 Family

- Single five volt supply
- N channel, silicon gate, depletion load technology
- Eight bit parallel processing
- 56 Instructions
- Decimal and binary arithmetic
- Thirteen addressing modes
- True indexing capability
- Programmable stack pointer
- Variable length stack
- Interrupt capability
- Non-maskable interrupt
- Use with any type or speed memory
- Bi-directional Data Bus
- Instruction decoding and control
- Addressable memory range of up to 65K bytes
- "Ready" input
- Direct memory access capability
- Bus compatible with MC6800
- Choice of external or on-board clocks
- 1MHz and 2MHz operation
- On-the-chip clock options
  * External single clock input
  * RC time base input
  * Crystal time base input
- 40 and 28 pin package versions
- Pipeline architecture

## Members of the Family

Microprocessors with On-Board Clock Oscillator

- MCS6502
- MCS6503
- MCS6504
- MCS6505
- MCS6506

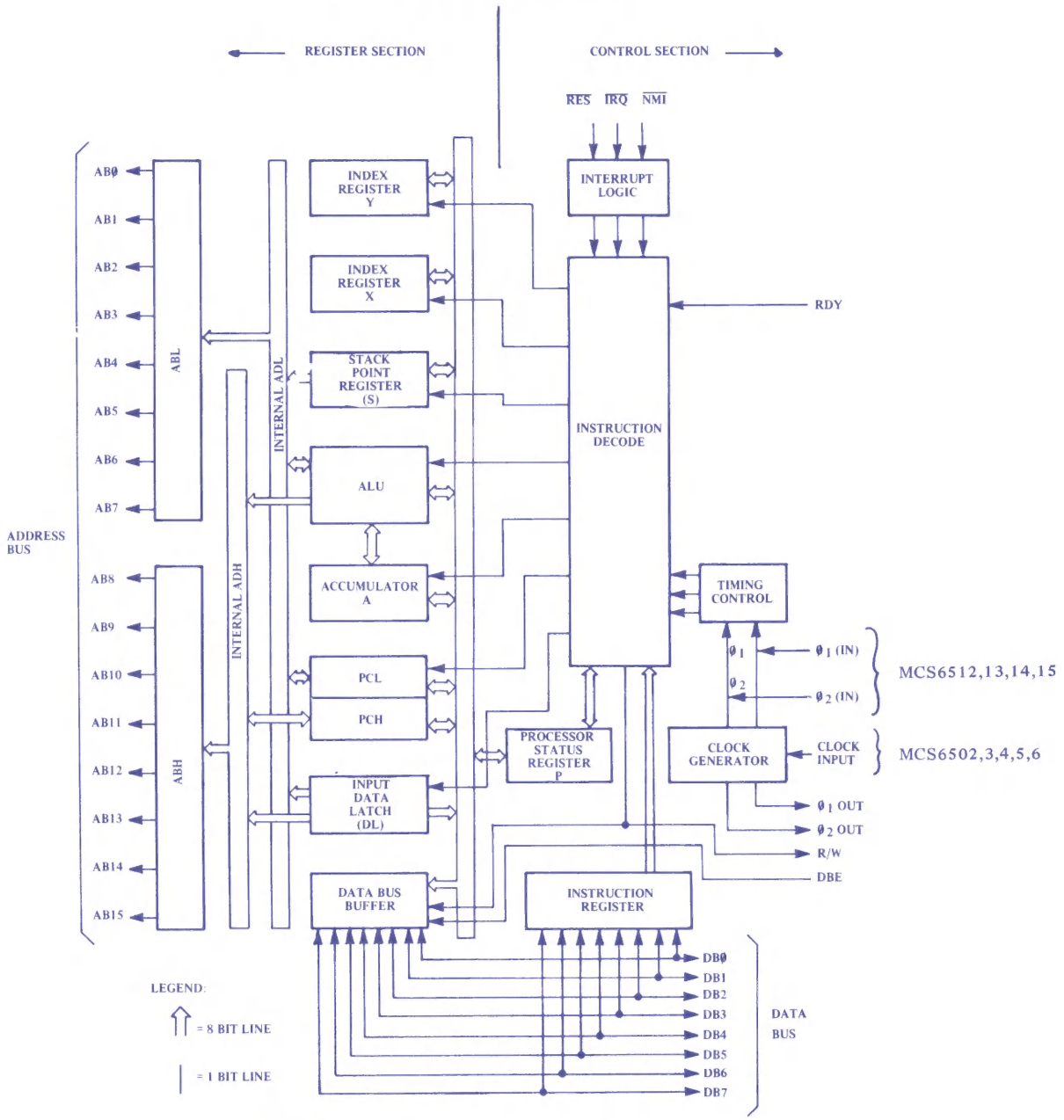
Microprocessors with External Two Phase Clock Input

- MCS6512
- MCS6513
- MCS6514
- MCS6515

## Comments on the Data Sheet

The data sheet is constructed to review first the basic "Common Characteristics" - those features which are common to the general family of microprocessors. Subsequent to a review of the family characteristics will be sections devoted to each member of the group with specific features of each.

## COMMON CHARACTERISTICS




Note: 1. Clock Generator is not included on MCS6512,13,14,15

2. Addressing Capability and control options vary with each of the MCS6500 Products.

**MCS6500 Internal Architecture**

## COMMON CHARACTERISTICS

### MAXIMUM RATINGS

| RATING | SYMBOL | VALUE | UNIT |
|---|---|---|---|
| SUPPLY VOLTAGE | Vcc | -0.3 to +7.0 | Vdc |
| INPUT VOLTAGE | Vin | -0.3 to +7.0 | Vdc |
| OPERATING TEMPERATURE | $T_A$ | 0 to +70 | °C |
| STORAGE TEMPERATURE | $T_{STG}$ | -55 to +150 | °C |

This device contains input protection against damage due to high static voltages or electric fields; however, precautions should be taken to avoid application of voltages higher than the maximum rating.

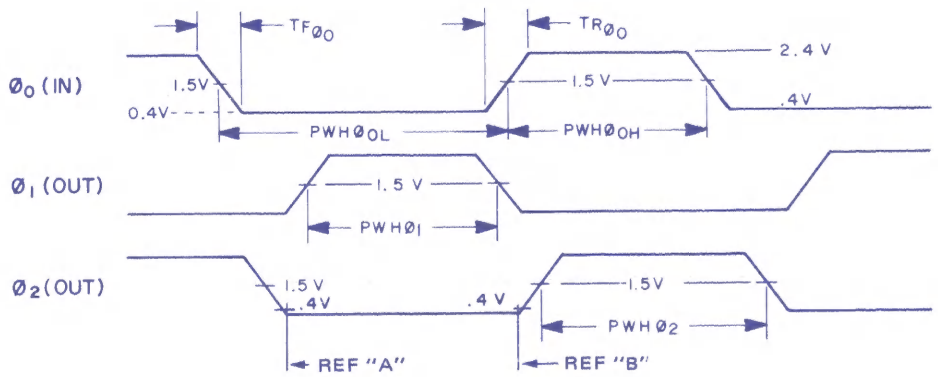
### ELECTRICAL CHARACTERISTICS (Vcc = 5.0V ± 5%, Vss = 0, $T_A$ = 25° C)

$\phi_1$, $\phi_2$ applies to MCS6512, 13, 14, 15, $\phi_{o\ (in)}$ applies to MCS6502, 03, 04, 05 and 06

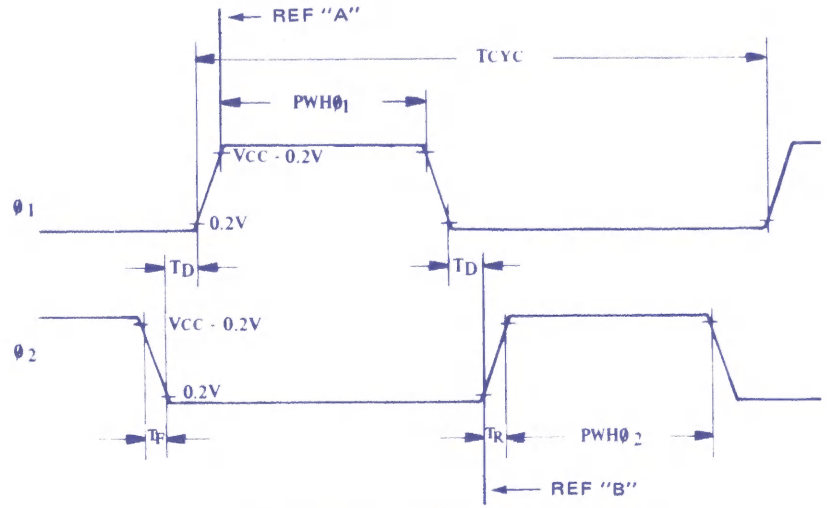
| CHARACTERISTIC | SYMBOL | MIN. | TYP. | MAX. | UNIT |
|---|---|---|---|---|---|
| Input High Voltage | $V_{IH}$ | | | | Vdc |
| Logic, $\phi_{o(in)}$ | | Vss + 2.4 | - | Vcc | |
| $\phi_1$, $\phi_2$ | | Vcc - 0.2 | - | Vcc + 0.25 | |
| Input Low Voltage | $V_{IL}$ | | | | Vdc |
| Logic, $\phi_{o(in)}$ | | Vss - 0.3 | - | Vss + 0.4 | |
| $\phi_1$, $\phi_2$ | | Vss - 0.3 | - | Vss + 0.2 | |
| Input High Threshold Voltage | $V_{IHT}$ | | | | |
| $\overline{RES}$, $\overline{NMI}$, RDY, $\overline{IRQ}$, Data, S.O. | | Vss + 2.0 | - | - | Vdc |
| Input Low Threshold Voltage | $V_{ILT}$ | | | | |
| $\overline{RES}$, $\overline{NMI}$, RDY, $\overline{IRQ}$, Data, S.O. | | - | - | Vss + 0.8 | Vdc |
| Input Leakage Current ($V_{in}$ = 0 to 5.25V, Vcc = 0) | $I_{in}$ | | | | |
| Logic (Excl. RDY, S.O.) | | - | - | 2.5 | μA |
| $\phi_1$, $\phi_2$ | | - | - | 100 | μA |
| $\phi_{o(in)}$ | | - | - | 10.0 | μA |
| Three-State (Off State) Input Current ($V_{in}$ = 0.4 to 2.4V, Vcc = 5.25V) | $I_{TSI}$ | | | | μA |
| Data Lines | | - | - | 10 | |
| Output High Voltage ($I_{LOAD}$ = -100μAdc, Vcc = 4.75V) | $V_{OH}$ | | | | |
| SYNC, Data, A0-A15, R/W | | Vss + 2.4 | - | - | Vdc |
| Output Low Voltage ($I_{LOAD}$ = 1.6mAdc, Vcc = 4.75V) | $V_{OL}$ | | | | |
| SYNC, Data, A0-A15, R/W | | - | - | Vss + 0.4 | Vdc |
| Power Dissipation | $P_D$ | - | .25 | .70 | W |
| Capacitance ($V_{in}$ = 0, $T_A$ = 25°C, f = 1MHz) | C | | | | pF |
| Logic | $C_{in}$ | - | - | 10 | |
| Data | | - | - | 15 | |
| A0-A15, R/W, SYNC | $C_{out}$ | - | - | 12 | |
| $\phi_{o(in)}$ | $C_{\phi_{o(in)}}$ | - | - | 15 | |
| $\phi_1$ | $C_{\phi_1}$ | - | 30 | 50 | |
| $\phi_2$ | $C_{\phi_2}$ | - | 50 | 80 | |

Note: $\overline{IRQ}$ and $\overline{NMI}$ require 3K pull-up resistors.

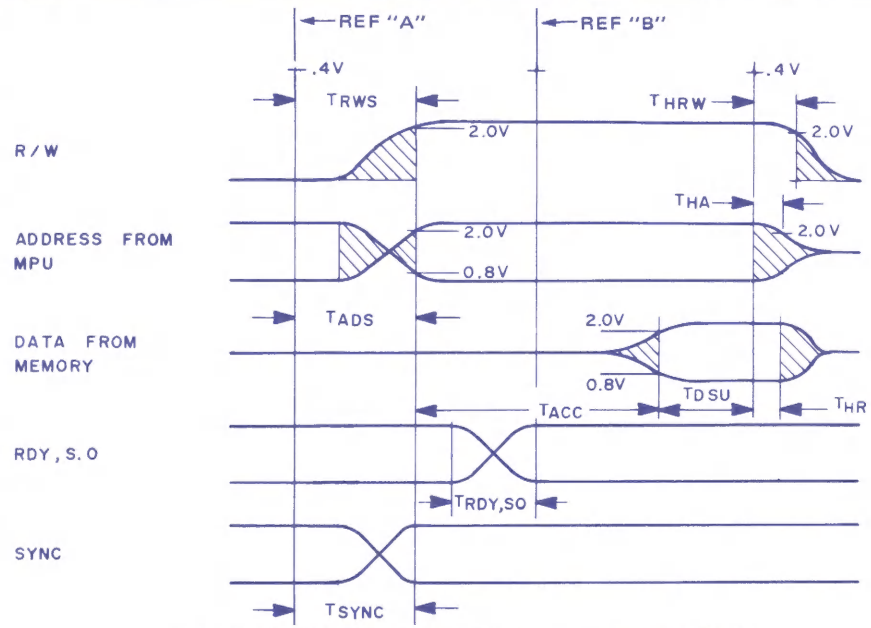
## COMMON CHARACTERISTICS



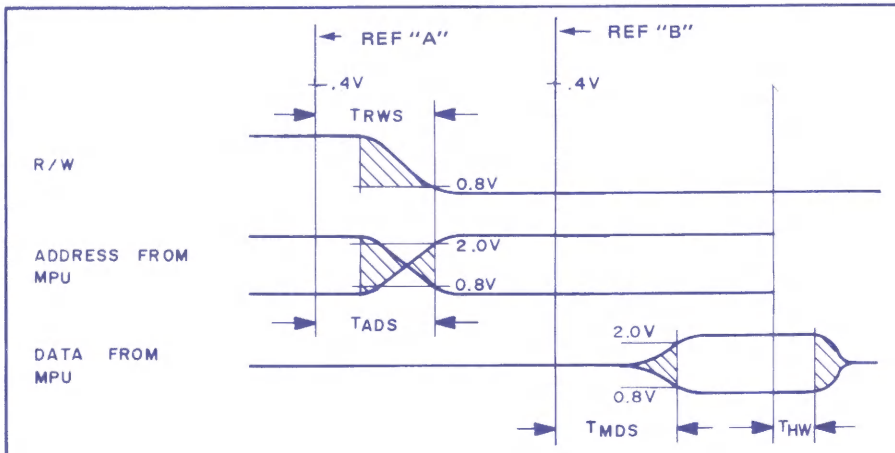

Clock Timing – MCS6502, 03, 04, 05, 06




Clock Timing – MCS6512, 13, 14, 15




Timing for Reading Data from Memory or Peripherals




Timing for Writing Data to Memory or Peripherals

Note: "REF." means Reference Points on clocks.

## 1 MHz TIMING

### Clock Timing – MCS6512, 13, 14, 15

| CHARACTERISTIC | SYMBOL | MIN. | TYP. | MAX. | UNIT |
|---|---|---|---|---|---|
| Cycle Time | $T_{CYC}$ | 1000 | --- | --- | nsec |
| Clock Pulse Width Ø1<br>(Measured at Vcc - 0.2v) Ø2 | PWH Ø1<br>PWH Ø2 | 430<br>470 | --- | --- | nsec |
| Fall Time<br>(Measured from 0.2v to Vcc - 0.2v) | $T_F$ | --- | --- | 25 | nsec |
| Delay Time between Clocks<br>(Measured at 0.2v) | $T_D$ | 0 | --- | --- | nsec |

### CLOCK TIMING -MCS6502, 03, 04, 05, 06

| CHARACTERISTIC | SYMBOL | MIN. | TYP. | MAX. | UNITS |
|---|---|---|---|---|---|
| Cycle Time | $T_{CYC}$ | 1000 | -- | -- | ns |
| $\phi_{o(IN)}$ Pulse Width (measured at 1.5V) | $PWH\phi_o$ | 460 | -- | 520 | ns |
| $\phi_{o(IN)}$ Rise, Fall Time | $TR\phi_o$,$TF\phi_o$ | -- | -- | 10 | ns |
| Delay Time Between Clocks (measured at 1.5V) | $T_D$ | 5 | -- | -- | ns |
| $\phi_{1(OUT)}$ Pulse Width (measured at 1.5V) | $PWH\phi_1$ | $PWH\phi_{oL}$-20 | -- | $PWH\phi_{oL}$ | ns |
| $\phi_{2(OUT)}$ Pulse Width (measured at 1.5V) | $PWH\phi_2$ | $PWH\phi_{oH}$-40 | -- | $PWH\phi_{oH}$-10 | ns |
| $\phi_{1(OUT)}$, $\phi_{2(OUT)}$ Rise, Fall Time (measured .8V to 2.0 V) (Load = 30pf + 1 TTL) | $T_R$, $T_F$ | -- | -- | 25 | ns |

### READ/WRITE TIMING

| CHARACTERISTIC | SYMBOL | MIN. | TYP. | MAX. | UNITS |
|---|---|---|---|---|---|
| Read/Write Setup Time from MCS6500 | $T_{RWS}$ | -- | 100 | 300 | ns |
| Address Setup Time from MCS6500 | $T_{ADS}$ | -- | 100 | 300 | ns |
| Memory Read Access Time | $T_{ACC}$ | -- | -- | 575 | ns |
| Data Stability Time Period | $T_{DSU}$ | 100 | -- | -- | ns |
| Data Hold Time - Read | $T_{HR}$ | 10 | -- | -- | ns |
| Data Hold Time - Write | $T_{HW}$ | 30 | 60 | -- | ns |
| Data Setup Time from MCS6500 | $T_{MDS}$ | -- | 150 | 200 | ns |
| RDY, S.O. Setup Time | $T_{RDY}$ | 100 | -- | -- | ns |
| SYNC Setup Time from MCS6500 | $T_{SYNC}$ | -- | -- | 350 | ns |
| Address Hold Time | $T_{HA}$ | 30 | 60 | -- | ns |
| R/W Hold Time | $T_{HRW}$ | 30 | 60 | -- | ns |

## 2 MHz TIMING

### Clock Timing – MCS6512, 13, 14, 15, 16

| CHARACTERISTIC | SYMBOL | MIN. | TYP. | MAX. | UNIT |
|---|---|---|---|---|---|
| Cycle Time | $T_{CYC}$ | 500 | --- | --- | nsec |
| Clock Pulse Width Ø1<br>(Measured at Vcc - 0.2v) Ø2 | PWH Ø1<br>PWH Ø2 | 215<br>235 | --- | --- | nsec |
| Fall Time<br>(Measured from 0.2v to Vcc - 0.2v) | $T_F$ | --- | --- | 12 | nsec |
| Delay Time between Clocks<br>(Measured at 0.2v) | $T_D$ | 0 | --- | --- | nsec |

### CLOCK TIMING - MCS6502, 03, 04, 05, 06

| CHARACTERISTIC | SYMBOL | MIN. | TYP. | MAX. | UNITS |
|---|---|---|---|---|---|
| Cycle Time | $T_{CYC}$ | 500 | -- | -- | ns |
| $\phi_{o(IN)}$ Pulse Width (measured at 1.5V) | $PWH\phi_o$ | 240 | -- | 260 | ns |
| $\phi_{o(IN)}$ Rise, Fall Time | $TR\phi_o$,$TF\phi_o$ | -- | -- | 10 | ns |
| Delay Time Between Clocks (measured at 1.5V) | $T_D$ | 5 | -- | -- | ns |
| $\phi_{1(OUT)}$ Pulse Width (measured at 1.5V) | $PWH\phi_1$ | $PWH\phi_{oL}$-20 | -- | $PWH\phi_{oL}$ | ns |
| $\phi_{2(OUT)}$ Pulse Width (measured at 1.5V) | $PWH\phi_2$ | $PWH\phi_{oH}$-40 | -- | $PWH\phi_{oH}$-10 | ns |
| $\phi_{1(OUT)}$, $\phi_{2(OUT)}$ Rise, Fall Time (measured .8V to 2.0 V) (Load = 30pf + 1 TTL) | $T_R$, $T_F$ | -- | -- | 25 | ns |

### READ/WRITE TIMING

| CHARACTERISTIC | SYMBOL | MIN. | TYP. | MAX. | UNITS |
|---|---|---|---|---|---|
| Read/Write Setup Time from MCS6500A | $T_{RWS}$ | -- | 100 | 150 | ns |
| Address Setup Time from MCS6500A | $T_{ADS}$ | -- | 100 | 150 | ns |
| Memory Read Access Time | $T_{ACC}$ | -- | -- | 300 | ns |
| Data Stability Time Period | $T_{DSU}$ | 50 | -- | -- | ns |
| Data Hold Time - Read | $T_{HR}$ | 10 | -- | -- | ns |
| Data Hold Time - Write | $T_{HW}$ | 30 | 60 | -- | ns |
| Data Setup Time from MCS6500A | $T_{MDS}$ | -- | 75 | 100 | ns |
| RDY, S.O. Setup Time | $T_{RDY}$ | 50 | -- | -- | ns |
| SYNC Setup Time from MCS6500A | $T_{SYNC}$ | -- | -- | 175 | ns |
| Address Hold Time | $T_{HA}$ | 30 | 60 | -- | ns |
| R/W Hold Time | $T_{HRW}$ | 30 | 60 | -- | ns |

# COMMON CHARACTERISTICS

Clocks ($\phi_1$, $\phi_2$)

The MCS651X requires a two phase non-overlapping clock that runs at the Vcc voltage level.

The MCS650X clocks are supplied with an internal clock generator. The frequency of these clocks is externally controlled. Details of this feature are discussed in the MCS6502 portion of this data sheet.

Address Bus ($A_0$-$A_{15}$) (See sections on each micro for respective address lines on those devices.)

These outputs are TTL compatible, capable of driving one standard TTL load and 130pf.

Data Bus ($D_0$-$D_7$)

Eight pins are used for the data bus. This is a bi-directional bus, transferring data to and from the device and peripherals. The outputs are tri-state buffers capable of driving one standard TTL load and 130pf.

Data Bus Enable (DBE)

This TTL compatible input allows external control of the tri-state data output buffers and will enable the microprocessor bus driver when in the high state. In normal operation DBE would be driven by the phase two ($\phi_2$) clock, thus allowing data output from microprocessor only during $\phi_2$. During the read cycle, the data bus drivers are internally disabled, becoming essentially an open circuit. To disable data bus drivers externally, DBE should be held low.

Ready (RDY)

This input signal allows the user to single cycle the microprocessor on all cycles except write cycles. A negative transition to the low state during or coincident with phase one ($\phi_1$) will halt the microprocessor with the output address lines reflecting the current address being fetched. This condition will remain through a subsequent phase two ($\phi_2$) in which the Ready signal is low. This feature allows microprocessor interfacing with low speed PROMS as well as fast (max. 2 cycle) Direct Memory Access (DMA). If Ready is low during a write cycle, it is ignored until the following read operation.

Interrupt Request ($\overline{IRQ}$)

This TTL level input requests that an interrupt sequence begin within the microprocessor. The microprocessor will complete the current instruction being executed before recognizing the request. At that time, the interrupt mask bit in the Status Code Register will be examined. If the interrupt mask flag is not set, the microprocessor will begin an interrupt sequence. The Program Counter and Processor Status Register are stored in the stack. The microprocessor will then set the interrupt mask flag high so that no further interrupts may occur. At the end of this cycle, the program counter low will be loaded from address FFFE, and program counter high from location FFFF, therefore transferring program control to the memory vector located at these addresses. The RDY signal must be in the high state for any interrupt to be recognized. A 3KΩ external resistor should be used for proper wire-OR operation.

Non-Maskable Interrupt ($\overline{NMI}$)

A negative going edge on this input requests that a non-maskable interrupt sequence be generated within the microprocessor.

$\overline{NMI}$ is an unconditional interrupt. Following completion of the current instruction, the sequence of operations defined for $\overline{IRQ}$ will be performed, regardless of the state interrupt mask flag. The vector address loaded into the program counter, low and high, are locations FFFA and FFFB respectively, thereby transferring program control to the memory vector located at these addresses. The instructions loaded at these locations cause the microprocessor to branch to a non-maskable interrupt routine in memory.

$\overline{NMI}$ also requires an external 3KΩ register to Vcc for proper wire-OR operations.

Inputs $\overline{IRQ}$ and $\overline{NMI}$ are hardware interrupts lines that are sampled during $\phi_2$ (phase 2) and will begin the appropriate interrupt routine on the $\phi_1$ (phase 1) following the completion of the current instruction.

Set Overflow Flag (S.O.)

A NEGATIVE going edge on this input sets the overflow bit in the Status Code Register. This signal is sampled on the trailing edge of $\phi_1$.

SYNC

This output line is provided to identify those cycles in which the microprocessor is doing an OP CODE fetch. The SYNC line goes high during $\phi_1$ of an OP CODE fetch and stays high for the remainder of that cycle. If the RDY line is pulled low during the $\phi_1$ clock pulse in which SYNC went high, the processor will stop in its current state and will remain in the state until the RDY line goes high. In this manner, the SYNC signal can be used to control RDY to cause single instruction execution.

Reset

This input is used to reset or start the microprocessor from a power down condition. During the time that this line is held low, writing to or from the microprocessor is inhibited. When a positive edge is detected on the input, the microprocessor will immediately begin the reset sequence.

After a system initialization time of six clock cycles, the mask interrupt flag will be set and the microprocessor will load the program counter from the memory vector locations FFFC and FFFD. This is the start location for program control.

After Vcc reaches 4.75 volts in a power up routine, reset must be held low for at least two clock cycles. At this time the R/W and (SYNC) signal will become valid.

When the reset signal goes high following these two clock cycles, the microprocessor will proceed with the normal reset procedure detailed above.

# COMMON CHARACTERISTICS

## INSTRUCTION SET – ALPHABETIC SEQUENCE

ADC Add Memory to Accumulator with Carry
AND "AND" Memory with Accumulator
ASL Shift left One Bit (Memory or Accumulator)

BCC Branch on Carry Clear
BCS Branch on Carry Set
BEQ Branch on Result Zero
BIT Test Bits in Memory with Accumulator
BMI Branch on Result Minus
BNE Branch on Result not Zero
BPL Branch on Result Plus
BRK Force Break
BVC Branch on Overflow Clear
BVS Branch on Overflow Set

CLC Clear Carry Flag
CLD Clear Decimal Mode
CLI Clear Interrupt Disable Bit
CLV Clear Overflow Flag
CMP Compare Memory and Accumulator
CPX Compare Memory and Index X
CPY Compare Memory and Index Y

DEC Decrement Memory by One
DEX Decrement Index X by One
DEY Decrement Index Y by One

EOR "Exclusive-or" Memory with Accumulator

INC Increment Memory by One
INX Increment Index X by One
INY Increment Index Y by One

JMP Jump to New Location
JSR Jump to New Location Saving Return Address

LDA Load Accumulator with Memory
LDX Load Index X with Memory
LDY Load Index Y with Memory
LSR Shift One Bit Right (Memory or Accumulator)

NOP No Operation

ORA "OR Memory with Accumulator

PHA Push Accumulator on Stack
PHP Push Processor Status on Stack
PLA Pull Accumulator from Stack
PLP Pull Processor Status from Stack

ROL Rotate One Bit Left (Memory or Accumulator)
ROR Rotate One Bit Right (Memory or Accumulator)
RTI Return from Interrupt
RTS Return from Subroutine

SBC Subtract Memory from Accumulator with Borrow
SEC Set Carry Flag
SED Set Decimal Mode
SEI Set Interrupt Disable Status
STA Store Accumulator in Memory
STX Store Index X in Memory
STY Store Index Y in Memory

TAX Transfer Accumulator to Index X
TAY Transfer Accumulator to Index Y
TSX Transfer Stack Pointer to Index X
TXA Transfer Index X to Accumulator
TXS Transfer Index X to Stack Pointer
TYA Transfer Index Y to Accumulator

## ADDRESSING MODES

ACCUMULATOR ADDRESSING - This form of addressing is represented with a one byte instruction, implying an operation on the accumulator.

IMMEDIATE ADDRESSING - In immediate addressing, the operand is contained in the second byte of the instruction, with no further memory addressing required.

ABSOLUTE ADDRESSING - In absolute addressing, the second byte of the instruction specifies the eight low order bits of the effective address while the third byte specifies the eight high order bits. Thus, the absolute addressing mode allows access to the entire 65K bytes of addressable memory.

ZERO PAGE ADDRESSING - The zero page instructions allow for shorter code and execution times by only fetching the second byte of the instruction and assuming a zero high address byte. Careful use of the zero page can result in significant increase in code efficiency.

INDEXED ZERO PAGE ADDRESSING - (X, Y indexing) - This form of addressing is used in conjunction with the index register and is referred to as "Zero Page, X" or "Zero Page, Y". The effective address is calculated by adding the second byte to the contents of the index register. Since this is a form of "Zero Page" addressing, the content of the second byte references a location in page zero. Additionally due to the "Zero Page" addressing nature of this mode, no carry is added to the high order 8 bits of memory and crossing of page boundaries does not occur.

INDEXED ABSOLUTE ADDRESSING - (X, Y indexing) - This form of addressing is used in conjunction with X and Y index register and is referred to as "Absolute, X", and "Absolute, Y". The effective address is formed by adding the contents of X or Y to the address contained in the second and third bytes of the instruction. This mode allows the index register to contain the index or count value and the instruction to contain the base address. This type of indexing allows any location referencing and the index to modify multiple fields resulting in reduced coding and execution time.

IMPLIED ADDRESSING - In the implied addressing mode, the address containing the operand is implicitly stated in the operation code of the instruction.

RELATIVE ADDRESSING - Relative addressing is used only with branch instructions and establishes a destination for the conditional branch.

The second byte of the instruction becomes the operand which is an "Offset" added to the contents of the lower eight bits of the program counter when the counter is set at the next instruction. The range of the offset is -128 to +127 bytes from the next instruction.

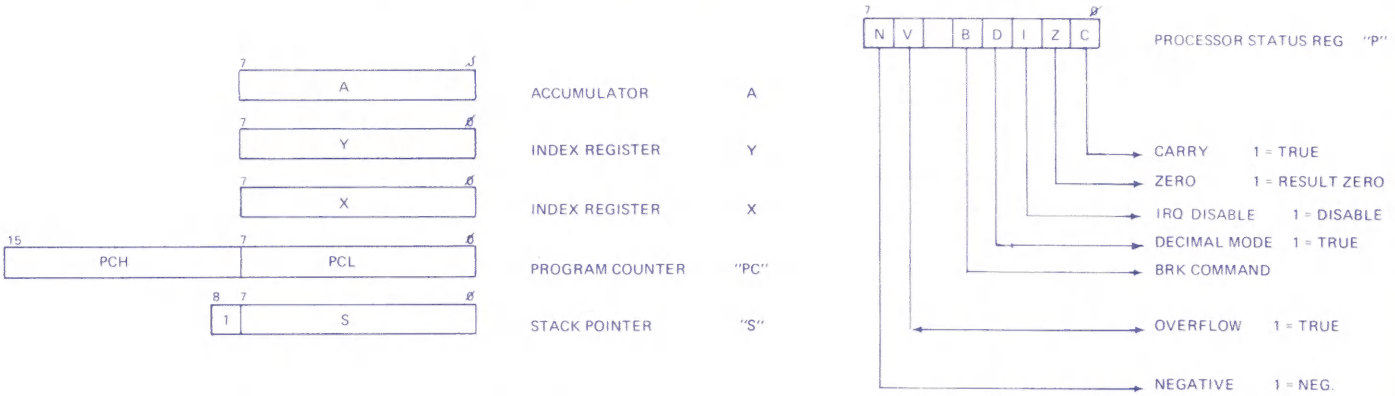
INDEXED INDIRECT ADDRESSING - In indexed indirect addressing (referred to as (Indirect,X)), the second byte of the instruction is added to the contents of the X index register, discarding the carry. The result of this addition points to a memory location on page zero whose contents is the low order eight bits of the effective address. The next memory location in page zero contains the high order eight bits of the effective address. Both memory locations specifying the high and low order bytes of the effective address must be in page zero.

INDIRECT INDEXED ADDRESSING - In indirect indexed addressing (referred to as (Indirect),Y), the second byte of the instruction points to a memory location in page zero. The contents of this memory location is added to the contents of the Y index register, the result being the low order eight bits of the effective address. The carry from this addition is added to the contents of the next page zero memory location, the result being the high order eight bits of the effective address.

ABSOLUTE INDIRECT - The second byte of the instruction contains the low order eight bits of a memory location. The high order eight bits of that memory location is contained in the third byte of the instruction. The contents of the fully specified memory location is the low order byte of the effective address. The next memory location contains the high order byte of the effective address which is loaded into the sixteen bits of the program counter.

# COMMON CHARACTERISTICS

## PROGRAMMING MODEL




## INSTRUCTION SET – OP CODES, Execution Time, Memory Requirements

| INSTRUCTIONS | | IMMEDIATE | | | ABSOLUTE | | | ZERO PAGE | | | ACCUM. | | | IMPLIED | | | (IND,X) | | | (IND),Y | | | Z,PAGE,X | | | ABS,X | | | ABS,Y | | | RELATIVE | | | INDIRECT | | | Z,PAGE,Y | | | CONDITION CODES | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| MNEMONIC | OPERATION | OP | N | # | OP | N | # | OP | N | # | OP | N | # | OP | N | # | OP | N | # | OP | N | # | OP | N | # | OP | N | # | OP | N | # | OP | N | # | OP | N | # | OP | N | # | N | Z | C | I | D | V |
| ADC | A+M+C → A (4) (1) | 69 | 2 | 2 | 6D | 4 | 3 | 65 | 3 | 2 | | | | | | | 61 | 6 | 2 | 71 | 5 | 2 | 75 | 4 | 2 | 7D | 4 | 3 | 79 | 4 | 3 | | | | | | | | | | ✓ | ✓ | ✓ | – | – | ✓ |
| AND | A∧M→A (1) | 29 | 2 | 2 | 2D | 4 | 3 | 25 | 3 | 2 | | | | | | | 21 | 6 | 2 | 31 | 5 | 2 | 35 | 4 | 2 | 3D | 4 | 3 | 39 | 4 | 3 | | | | | | | | | | ✓ | ✓ | – | – | – | – |
| ASL | C 7 0 ← 0 | | | | 0E | 6 | 3 | 06 | 5 | 2 | 0A | 2 | 1 | | | | | | | | | | 16 | 6 | 2 | 1E | 7 | 3 | | | | | | | | | | | | | ✓ | ✓ | ✓ | – | – | – |
| BCC | BRANCH ON C=0 (2) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 90 | 2 | 2 | | | | | | | – | – | – | – | – | – |
| BCS | BRANCH ON C=1 (2) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | B0 | 2 | 2 | | | | | | | – | – | – | – | – | – |
| BEQ | BRANCH ON Z=1 (2) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | F0 | 2 | 2 | | | | | | | – | – | – | – | – | – |
| BIT | A∧M | | | | 2C | 4 | 3 | 24 | 3 | 2 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | $M_7$ | ✓ | – | – | $M_6$ |
| BMI | BRANCH ON N=1 (2) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 30 | 2 | 2 | | | | | | | – | – | – | – | – | – |
| BNE | BRANCH ON Z=0 (2) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | D0 | 2 | 2 | | | | | | | – | – | – | – | – | – |
| BPL | BRANCH ON N=0 (2) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 10 | 2 | 2 | | | | | | | – | – | – | – | – | – |
| BRK | (See Fig. 1) | | | | | | | | | | | | | 00 | 7 | 1 | | | | | | | | | | | | | | | | | | | | | | | | | – | – | – | 1 | – | – |
| BVC | BRANCH ON V=0 (2) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 50 | 2 | 2 | | | | | | | – | – | – | – | – | – |
| BVS | BRANCH ON V=1 (2) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 70 | 2 | 2 | | | | | | | – | – | – | – | – | – |
| CLC | 0 → C | | | | | | | | | | | | | 18 | 2 | 1 | | | | | | | | | | | | | | | | | | | | | | | | | – | – | – | 0 | – | – |
| CLD | 0 → D | | | | | | | | | | | | | D8 | 2 | 1 | | | | | | | | | | | | | | | | | | | | | | | | | – | – | – | – | 0 | – |
| CLI | 0 → I | | | | | | | | | | | | | 58 | 2 | 1 | | | | | | | | | | | | | | | | | | | | | | | | | – | – | – | 0 | – | – |
| CLV | 0 → V | | | | | | | | | | | | | B8 | 2 | 1 | | | | | | | | | | | | | | | | | | | | | | | | | – | – | – | – | – | 0 |
| CMP | A–M (1) | C9 | 2 | 2 | CD | 4 | 3 | C5 | 3 | 2 | | | | | | | C1 | 6 | 2 | D1 | 5 | 2 | D5 | 4 | 2 | DD | 4 | 3 | D9 | 4 | 3 | | | | | | | | | | ✓ | ✓ | ✓ | – | – | – |
| CPX | X–M | E0 | 2 | 2 | EC | 4 | 3 | E4 | 3 | 2 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | ✓ | ✓ | ✓ | – | – | – |
| CPY | Y–M | C0 | 2 | 2 | CC | 4 | 3 | C4 | 3 | 2 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | ✓ | ✓ | ✓ | – | – | – |
| DEC | M–1 → M | | | | CE | 6 | 3 | C6 | 5 | 2 | | | | | | | | | | | | | D6 | 6 | 2 | DE | 7 | 3 | | | | | | | | | | | | | ✓ | ✓ | – | – | – | – |
| DEX | X–1 → X | | | | | | | | | | | | | CA | | | | | | | | | | | | | | | | | | | | | | | | | | | ✓ | ✓ | – | – | – | – |
| DEY | Y–1 → Y | | | | | | | | | | | | | 88 | | | | | | | | | | | | | | | | | | | | | | | | | | | ✓ | ✓ | – | – | – | – |
| EOR | A ⊻ M → A (1) | 49 | 2 | 2 | 4D | 4 | 3 | 45 | 3 | 2 | | | | | | | 41 | 6 | 2 | 51 | 5 | 2 | 55 | 4 | 2 | 5D | 4 | 3 | 59 | 4 | 3 | | | | | | | | | | ✓ | ✓ | – | – | – | – |
| INC | M+1 → M | | | | EE | 6 | 3 | E6 | 5 | 2 | | | | | | | | | | | | | F6 | 6 | 2 | FE | 7 | 3 | | | | | | | | | | | | | ✓ | ✓ | – | – | – | – |
| INX | X+1 → X | | | | | | | | | | | | | E8 | 2 | 1 | | | | | | | | | | | | | | | | | | | | | | | | | ✓ | ✓ | – | – | – | – |
| INY | Y+1 → Y | | | | | | | | | | | | | C8 | 2 | 1 | | | | | | | | | | | | | | | | | | | | | | | | | ✓ | ✓ | – | – | – | – |
| JMP | JUMP TO NEW LOC. | | | | 4C | 3 | 3 | | | | | | | | | | | | | | | | | | | | | | | | | | | | 6C | 5 | 3 | | | | – | – | – | – | – | – |
| JSR | (See Fig. 2) JUMP SUB | | | | 20 | 6 | 3 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | – | – | – | – | – | – |
| LDA | M → A (1) | A9 | 2 | 2 | AD | 4 | 3 | A5 | 3 | 2 | | | | | | | A1 | 6 | 2 | B1 | 5 | 2 | B5 | 4 | 2 | BD | 4 | 3 | B9 | 4 | 3 | | | | | | | | | | ✓ | ✓ | – | – | – | – |

| MNEMONIC | OPERATION | IMMEDIATE | | | ABSOLUTE | | | ZERO PAGE | | | ACCUM. | | | IMPLIED | | | (IND,X) | | | (IND),Y | | | Z,PAGE,X | | | ABS,X | | | ABS,Y | | | RELATIVE | | | INDIRECT | | | Z,PAGE,Y | | | CONDITION CODES | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | OP | N | # | OP | N | # | OP | N | # | OP | N | # | OP | N | # | OP | N | # | OP | N | # | OP | N | # | OP | N | # | OP | N | # | OP | N | # | OP | N | # | OP | N | # | N | Z | C | I | D | V |
| LDX | M → X (1) | A2 | 2 | 2 | AE | 4 | 3 | A6 | 3 | 2 | | | | | | | | | | | | | | | | | | | BE | 4 | 3 | | | | | | | | | | B6 | 4 | 2 | ✓ | ✓ | – | – | – | – |
| LDY | M → Y (1) | A0 | 2 | 2 | AC | 4 | 3 | A4 | 3 | 2 | | | | | | | | | | | | | B4 | 4 | 2 | BC | 4 | 3 | | | | | | | | | | | | | ✓ | ✓ | – | – | – | – |
| LSR | 0 → 7 → 0 → C | | | | 4E | 6 | 3 | 46 | 5 | 2 | 4A | 2 | 1 | | | | | | | | | | 56 | 6 | 2 | 5E | 7 | 3 | | | | | | | | | | | | | 0 | ✓ | ✓ | – | – | – |
| NOP | NO OPERATION | | | | | | | | | | | | | EA | 2 | 1 | | | | | | | | | | | | | | | | | | | | | | | | | – | – | – | – | – | – |
| ORA | A∨M → A | 09 | 2 | 2 | 0D | 4 | 3 | 05 | 3 | 2 | | | | | | | 01 | 6 | 2 | 11 | 5 | 2 | 15 | 4 | 2 | 1D | 4 | 3 | 19 | 4 | 3 | | | | | | | | | | ✓ | ✓ | – | – | – | – |
| PHA | A → $M_S$ S–1 → S | | | | | | | | | | | | | 48 | 3 | 1 | | | | | | | | | | | | | | | | | | | | | | | | | – | – | – | – | – | – |
| PHP | P → $M_S$ S–1 → S | | | | | | | | | | | | | 08 | 3 | 1 | | | | | | | | | | | | | | | | | | | | | | | | | – | – | – | – | – | – |
| PLA | S+1 → S $M_S$ → A | | | | | | | | | | | | | 68 | 4 | 1 | | | | | | | | | | | | | | | | | | | | | | | | | ✓ | ✓ | – | – | – | – |
| PLP | S+1 → S $M_S$ → P | | | | | | | | | | | | | 28 | 4 | 1 | | | | | | | | | | | | | | | | | | | | | | | | | (RESTORED) | | | | | |
| ROL | C ← 7 ← 0 ← C | | | | 2E | 6 | 3 | 26 | 5 | 2 | 2A | 2 | 1 | | | | | | | | | | 36 | 6 | 2 | 3E | 7 | 3 | | | | | | | | | | | | | ✓ | ✓ | ✓ | – | – | – |
| ROR | C → 7 → 0 → C | | | | 6E | 6 | 3 | 66 | 5 | 2 | 6A | 2 | 1 | | | | | | | | | | 76 | 6 | 2 | 7E | 7 | 3 | | | | | | | | | | | | | ✓ | ✓ | ✓ | – | – | – |
| RTI | (See Fig. 1) RTRN. INT. | | | | | | | | | | | | | 40 | 6 | 1 | | | | | | | | | | | | | | | | | | | | | | | | | (RESTORED) | | | | | |
| RTS | (See Fig. 2) RTRN SUB | | | | | | | | | | | | | 60 | 6 | 1 | | | | | | | | | | | | | | | | | | | | | | | | | – | – | – | – | – | – |
| SBC | A–M–$\bar{C}$ → A (1) | E9 | 2 | 2 | ED | 4 | 3 | E5 | 3 | 2 | | | | | | | E1 | 6 | 2 | F1 | 5 | 2 | F5 | 4 | 2 | FD | 4 | 3 | F9 | 4 | 3 | | | | | | | | | | ✓ | ✓ | (3) | – | – | ✓ |
| SEC | 1 → C | | | | | | | | | | | | | 38 | 2 | 1 | | | | | | | | | | | | | | | | | | | | | | | | | – | – | 1 | – | – | – |
| SED | 1 → D | | | | | | | | | | | | | F8 | 2 | 1 | | | | | | | | | | | | | | | | | | | | | | | | | – | – | – | – | 1 | – |
| SEI | 1 → I | | | | | | | | | | | | | 78 | 2 | 1 | | | | | | | | | | | | | | | | | | | | | | | | | – | – | – | 1 | – | – |
| STA | A → M | | | | 8D | 4 | 3 | 85 | 3 | 2 | | | | | | | 81 | 6 | 2 | 91 | 6 | 2 | 95 | 4 | 2 | 9D | 5 | 3 | 99 | 5 | 3 | | | | | | | | | | – | – | – | – | – | – |
| STX | X → M | | | | 8E | 4 | 3 | 86 | 3 | 2 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 96 | 4 | 2 | – | – | – | – | – | – |
| STY | Y → M | | | | 8C | 4 | 3 | 84 | 3 | 2 | | | | | | | | | | | | | 94 | 4 | 2 | | | | | | | | | | | | | | | | – | – | – | – | – | – |
| TAX | A → X | | | | | | | | | | | | | AA | 2 | 1 | | | | | | | | | | | | | | | | | | | | | | | | | ✓ | ✓ | – | – | – | – |
| TAY | A → Y | | | | | | | | | | | | | A8 | 2 | 1 | | | | | | | | | | | | | | | | | | | | | | | | | ✓ | ✓ | – | – | – | – |
| TSX | S → X | | | | | | | | | | | | | BA | 2 | 1 | | | | | | | | | | | | | | | | | | | | | | | | | ✓ | ✓ | – | – | – | – |
| TXA | X → A | | | | | | | | | | | | | 8A | 2 | 1 | | | | | | | | | | | | | | | | | | | | | | | | | ✓ | ✓ | – | – | – | – |
| TXS | X → S | | | | | | | | | | | | | 9A | 2 | 1 | | | | | | | | | | | | | | | | | | | | | | | | | – | – | – | – | – | – |
| TYA | Y → A | | | | | | | | | | | | | 98 | 2 | 1 | | | | | | | | | | | | | | | | | | | | | | | | | ✓ | ✓ | – | – | – | – |

(1) ADD 1 TO "N" IF PAGE BOUNDRY IS CROSSED

(2) ADD 1 TO "N" IF BRANCH OCCURS TO SAME PAGE.
ADD 2 TO "N" IF BRANCH OCCURS TO DIFFERENT PAGE.

(3) CARRY NOT = BORROW.

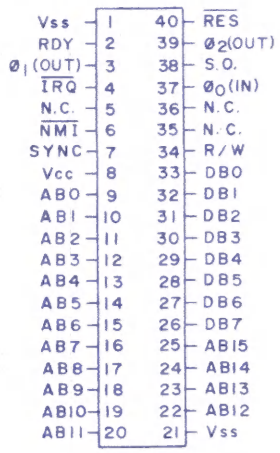
(4) IF IN DECIMAL MODE Z FLAG IS INVALID.
ACCUMULATOR MUST BE CHECKED FOR ZERO RESULT.

X INDEX . X
Y INDEX . Y
A ACCUMULATOR
M MEMORY PER EFFECTIVE ADDRESS
$M_S$ MEMORY PER STACK POINTER

\+ ADD
– SUBTRACT
∧ AND
∨ OR
⊻ EXCLUSIVE OR
✓ MODIFIED
– NOT MODIFIED
$M_7$ MEMORY BIT 7
$M_6$ MEMORY BIT 6

N NO. CYCLES
# NO. BYTES

## MCS6502 – 40 Pin Package

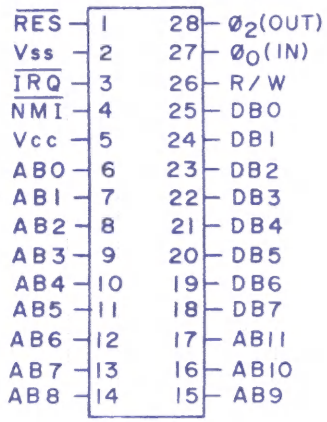



MCS6502

* 65K Addressable Bytes of Memory
* $\overline{IRQ}$ Interrupt
* $\overline{NMI}$ Interrupt
* On-the-chip Clock
  - ✓ TTL Level Single Phase Input
  - ✓ RC Time Base Input
  - ✓ Crystal Time Base Input
* SYNC Signal
  (can be used for single instruction execution)
* RDY Signal
  (can be used for single cycle execution)
* Two Phase Output Clock for Timing of Support Chips

**Features of MCS6502**

## MCS6503 – 28 Pin Package




MCS6503

* 4K Addressable Bytes of Memory (AB00-AB11)
* On-the-chip Clock
* $\overline{IRQ}$ Interrupt
* $\overline{NMI}$ Interrupt
* 8 Bit Bi-Directional Data Bus

**Features of MCS6503**

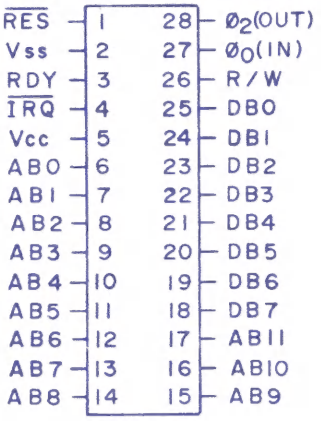
## MCS6504 – 28 Pin Package


MCS6504

* 8K Addressable Bytes of Memory (AB00-AB12)
* On-the-chip Clock
* $\overline{IRQ}$ Interrupt
* 8 Bit Bi-Directional Data Bus

**Features of MCS6504**

## MCS6505 – 28 Pin Package




MCS6505

* 4K Addressable Bytes of Memory (AB00-AB11)
* On-the-chip Clock
* $\overline{IRQ}$ Interrupt
* RDY Signal
* 8 Bit Bi-Directional Data Bus

**Features of MCS6505**

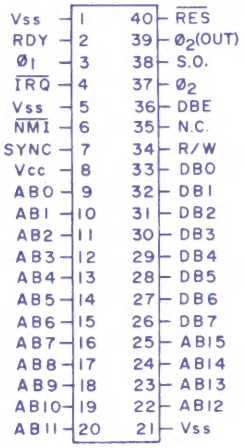
## MCS6506 – 28 Pin Package


MCS6506

* 4K Addressable Bytes of Memory (AB00-AB11)
* On-the-chip Clock
* $\overline{IRQ}$ Interrupt
* Two phases off
* 8 Bit Bi-Directional Data Bus

**Features of MCS6506**

## MCS6512 – 40 Pin Package




MCS6512

* 65K Addressable Bytes of Memory
* $\overline{IRQ}$ Interrupt
* $\overline{NMI}$ Interrupt
* RDY Signal
* 8 Bit Bi-Directional Data Bus
* SYNC Signal
* Two phase input
* Data Bus Enable

**Features of MCS6512**

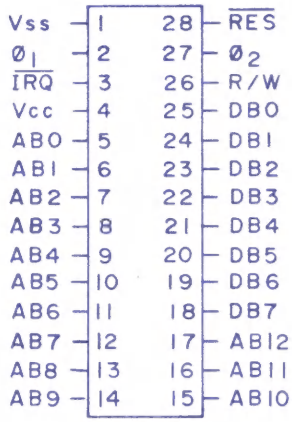
## MCS6513 – 28 Pin Package


MCS6513

* 4K Addressable Bytes of Memory (AB00-AB11)
* Two phase clock input
* $\overline{IRQ}$ Interrupt
* $\overline{NMI}$ Interrupt
* 8 Bit Bi-Directional Data Bus

**Features of MCS6513**

## MCS6514 – 28 Pin Package

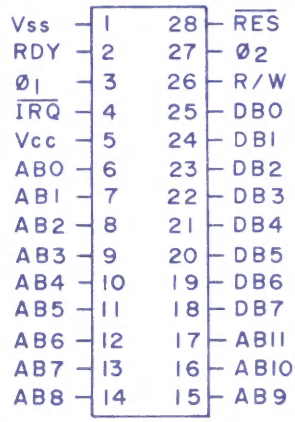



MCS6514

* 8K Addressable Bytes of Memory (AB00-AB12)
* Two phase clock input
* $\overline{IRQ}$ Interrupt
* 8 Bit Bi-Directional Data Bus

**Features of MCS6514**

## MCS6515 – 28 Pin Package



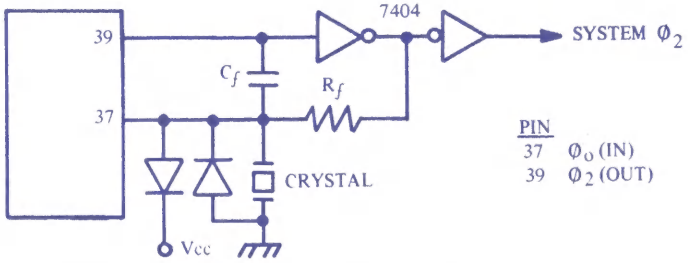

MCS6515

* 4K Addressable Bytes of Memory (AB00-AB11)
* Two phase clock input
* $\overline{IRQ}$ Interrupt
* 8 Bit Bi-Directional Data Bus

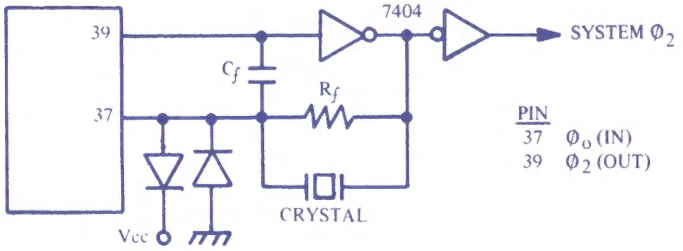
**Features of MCS6515**

# TIME BASE GENERATION OF INPUT CLOCK

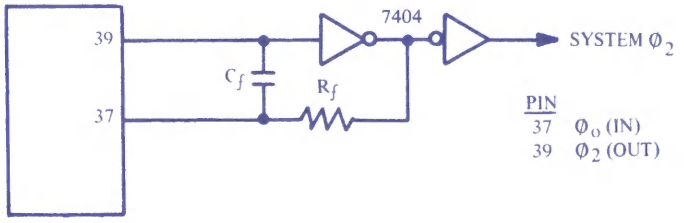
## MCS6502




*MCS6502 Parallel Mode Crystal Controlled Oscillator*

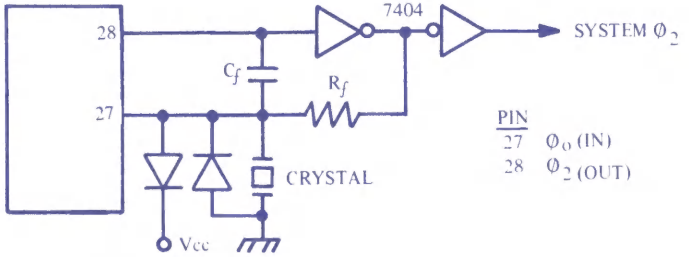



*MCS6502 Series Mode Crystal Controlled Oscillator*

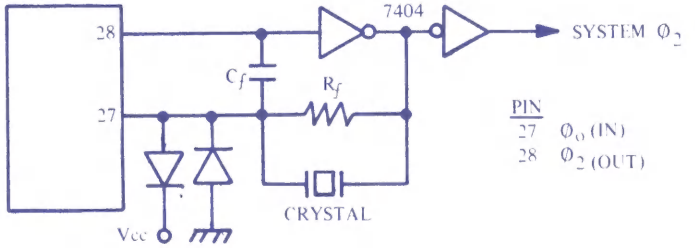



*MCS6502 Time Base Generator – RC Network*

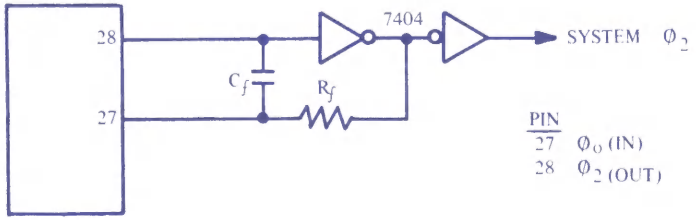
## MCS6503, MCS6504, MCS6505, MCS6506




*MCS6503,4,5,6 Parallel Mode Crystal Controlled Oscillator*




*MCS6503,4,5,6 Series Mode Crystal Controlled Oscillator*




*MCS6503, MCS6504, MCS6505, MCS6506 Time Base Generation RC Network*

brought to you by
andy finkel